On the false positives and false negatives of the Jacobian Matrix in kinematically redundant parallel mechanisms

Article (Accepted Version)

On the False Positives and False Negatives of the Jacobian Matrix in Kinematically Redundant Parallel Mechanisms

Nicholas Baron, *Student Member, IEEE,*
Andrew Philippides, *Member, IEEE,*
and Nicolas Rojas, *Member, IEEE*

***Abstract*—The Jacobian matrix is a highly popular tool for the control and performance analysis of closed-loop robots. Its usefulness in parallel mechanisms is certainly apparent, and its application to solve motion planning problems, or other higher level questions, has been seldom queried, or limited to non-redundant systems. In this paper, we discuss the shortcomings of the use of the Jacobian matrix under redundancy, in particular when applied to kinematically redundant parallel architectures with non-serially connected actuators. These architectures have become fairly popular recently as they allow the end-effector to achieve full rotations, which is an impossible task with traditional topologies. The problems with the Jacobian matrix in these novel systems arise from the need to eliminate redundant variables when forming it, resulting in both situations where the Jacobian incorrectly identifies singularities (*false positive*), and where it fails to identify singularities (*false negative*). These issues have thus far remained unaddressed in the literature. We highlight these limitations herein by demonstrating several cases using numerical examples of both planar and spatial architectures.**

I. Introduction

The Jacobian matrix is a well-renowned and popular tool used in the control and analysis of parallel robots, relating the output velocity of the manipulator's end-effector, or platform, with the input joint velocities. An important role that the Jacobian plays in the analysis of these robots is the identification of singularities [1], which are problematic configurations in which the total number of degrees of freedom of the mechanism changes, possibly resulting in a loss of control of the robot. The singularities of non-redundant parallel mechanisms were first addressed in [2], where the input and output velocity vectors, $\dot{\theta}$ and $\dot{\mathbf{x}}$, are related by the Jacobian matrices $\mathbf{A}$ and $\mathbf{B}$, such that $\mathbf{A}\dot{\mathbf{x}} + \mathbf{B}\dot{\theta} = 0$. Three singularity types were defined when each and both of $\mathbf{A}$ and $\mathbf{B}$ are singular. A more comprehensive singularity classification for non-redundant parallel robots was presented in [3], where six different types of singularities in parallel robots are defined, taking into account the passive joint velocities of the mechanism as well as the input and output variables. Furthermore, an additional category, termed the *constraint singularity*, is addressed in [4], which may occur when the Jacobian matrices describing the input-output relationship of a non-redundant parallel manipulator are non-singular.

For redundant parallel mechanisms, which are robotic systems where the number of input variables exceeds the dimension of the task space, the literature is much less comprehensive. Redundancy has been traditionally included in closed-loop manipulators to overcome one of the major limitations of these robots, namely their limited workspaces and rotational capabilities due to the existence of singularities [5], [6]; it was indeed a highly active research topic in the late 1990s and early 2000s (e.g., [7], [8]). The use of redundancy in parallel robots has regained popularity in recent years as it is useful for obtaining architectures able to complete full rotations of the end-effector [9], [10]. Just as for non-redundant robots, the Jacobian is frequently used to perform the singularity analysis of redundant parallel mechanisms. For instance, in [7], a solution to finding the singular configurations of redundantly actuated robots, for which the number of input parameters, n, exceeds the number of degrees of freedom of the platform, m, was proposed by generating a non-square $m \times n$ Jacobian, $\mathbf{J}$, such that $\dot{\mathbf{x}} = \mathbf{J}\dot{\theta}$, where the configuration is singular if $det(\mathbf{J}\mathbf{J}^T) = 0$. A similar analysis was conducted in [11], using singular value decomposition of the Jacobian, where three different singularity conditions were found.

It can be said that there exists two types of redundant parallel mechanisms: those that are redundantly actuated, and those that are kinematically redundant [9]. Kinematically redundant architectures correspond to mechanisms for which the number of actuated joints exceeds the task space of the end-effector, but all of which are required to be locked in order for the entire system to become rigid. This is in contrast to redundantly actuated mechanisms, in which not all actuators need to be active to fix the position of the end-effector with respect to the base. The advantage of kinematically redundant architectures is that they do not suffer from the generation of internal forces or moments due to the mechanism being over-constrained. There are two different ways to obtain kinematically redundant architectures; the first is to take a non-redundant architecture and add extra actuated joints in the kinematic chains (limbs) connecting the end-effector to the base, such that there now exists actuated joints connected in series [12]. The second is to develop architectures where two or more of the limbs are interconnected between the base and the platform such that none of the actuators are connected in series [13], [10]. We refer to these two types of kinematically redundant architectures as those with serially connected actuators and those with non-serially connected actuators, respectively. Herein, our focus is directed at the latter category of architectures, as it has been shown that some of these systems have exhibited unlimited rotational capabilities (*e.g.*, [9]), an impossible task with other parallel architectures, and have become fairly popular recently.

Several kinematically redundant architectures with non-serially connected actuators have been indeed proposed in the literature in the last five years. When conducting the kinematic analysis of these mechanisms, it is commonplace to formulate the so-called forward kinematic Jacobian by eliminating the additional passive joint velocity which describes the kinematically redundant degree of freedom—see for instance [9], [13], [14]. However, there are problems associated with this method which, to the authors' knowledge, have yet to be discussed in the literature. Such problems arise when using Jacobian-based methods of singularity analysis on these mechanisms. Herein, the term singularity is used to describe the so-called forward kinematics singularity, also referred to as a singularity of redundant output in [15], which describes a configuration where a non-zero output velocity vector is generated even when the actuators are locked. In this paper, the Jacobians of three kinematically redundant parallel robots with non-serially connected actuators (two planar and one spatial) are calculated following the standard approach, and particular instances of each architecture are examined. It is shown that the inverse of the 2-norm condition number of the Jacobian, a traditional method of singularity analysis, either fails to identify or incorrectly identifies a singular configuration. Indeed, other measures, such as computing the determinant of the Jacobian, exhibit the same shortcomings. This phenomenon is distinct from the constraint singularity, in that the Jacobian is either failing to identify (false negative), or incorrectly identifying (false positive), direct kinematic singularities. The failure of the Jacobian is verified using the principles of rigidity theory; by analysing the underlying graph of the robot and computing its rigidity matrix.

The rest of this paper is structured as follows. In section II, a summary of the principles of rigidity theory is given, including how a parallel robot can be analysed in terms of its underlying graph and

N. Baron and A. Philippides are with the School of Engineering and Informatics, University of Sussex, Brighton BN1 9RH, UK (e-mail: {n.baron, andrewop}@sussex.ac.uk)

N. Rojas is with the Dyson School of Design Engineering, Imperial College London, London SW7 2DB, UK (e-mail: n.rojas@imperial.ac.uk).

how its rigidity, or lack thereof, can be determined by computing the rank of its rigidity matrix. In section III, a family of kinematically redundant parallel mechanisms with non-serially connected actuators is presented, and in section IV, the methods used to calculate the Jacobian for these mechanisms are presented. In section V, three example configurations of these architectures where the Jacobian fails as a means of singularity analysis are demonstrated. In section VI the results are discussed and, finally, in section VII we conclude.

II. The Rigidity Matrix of Parallel robots

Rigidity theory provides a useful set of mathematical tools which can be leveraged for the analysis of parallel robots [16], [17], [18]. A graph $G = (V, E)$ is a set of $|V|$ vertices and $|E|$ edges, where each edge joins two vertices and is associated with a real number. A realisation of a graph is an assignment of coordinates to each vertex such that the Euclidean distances between any adjacent vertices equals the number associated with the corresponding edge. A framework, denoted by $p(G)$, is the combination of a graph and a realisation. A framework that can be continuously deformed whilst maintaining all of the distance constraints between the vertices is *flexible*, else it is *rigid* [16]. Parallel robots can be analysed using these principles through modelling the joints of the mechanism as the vertices of a graph and the links that join them as the edges [19]. A given configuration of a particular architecture can then be described as a framework, and the rigidity of the physical mechanism can be analysed by inspecting this framework.

A finite flexing of a framework $p(G)$ is defined as a family of realisations of G such that if the position of each vertex is differentiable with respect to time, the distance constraint $(p_i(t) - p_j(t))^2 = constant$ holds for each vertex pairing $(i, j) \in E$, and differentiating leads to

$$(v_i - v_j) \cdot (p_i - p_j) = 0 \tag{1}$$

where v_i is the instantaneous velocity of vertex i. An infinitesimal motion of a framework is a set of vertex velocities for which (1) holds for every paring of adjacent vertices; for generic graph realisations, infinitesimal motions correspond to finite flexings [16]. Finite flexings can be categorised as either trivial or non-trivial. Trivial finite flexings correspond to translations or rotations of the Euclidean space itself, non-trivial finite flexings are those that do not fit this description. If there exists a non-trivial infinitesimal motion, the framework is described as flexible, otherwise it is described as rigid.

In d-dimensional Euclidean space, a set of n vertices have nd possible independent motions. A d-dimensional body has d possible translations and $d(d-1)/2$ rotations, whereas a d'-dimensional body for which $d' < d$ has $d'(2d - d' - 1)/2$ rotations. The total number of allowed motions, $S(n, d)$, for the framework is given by the total number of independent motions of the vertices, nd, minus the number of rigid body motions, this is formulated by

$$S(n,d) = \begin{cases} nd - d(d+1)/2 & \text{if } n >= d, \\ n(n-1)/2 & \text{otherwise.} \end{cases} \tag{2}$$

If each edge adds one independent constraint, then $S(n, d)$ edges are required for the system to become rigid.

A method of testing the rigidity of a graph is by forming its rigidity matrix, which is comprised of the set of equations (1) for each edge. The matrix has m rows, each of which corresponds to an edge and nd columns, each of which corresponds to a coordinate of a vertex. If an element in the matrix is in a row corresponding to an edge and in a column of a vertex that is part of that edge, then the value of that element is the difference between that vertex and the other vertex in the edge in terms of the coordinates dictated by the column.

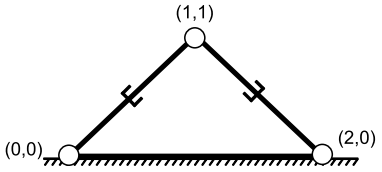

Fig. 1. Mechanism whose graph corresponds to three vertices, positioned at $P_1 = (0,0)^T$, $P_2 = (2,0)^T$, and $P_3 = (1,1)^T$, and three edges.

For example, consider the planar case depicted in Fig. 1 where two prismatic actuators are joined together by a revolute joint and also to the ground via two other revolute joints; assuming the actuators are locked, this mechanism can be modelled by a graph composed of three vertices, located at the positions of the revolute joints, joined by three edges, which correspond to the length between each pair of joints. For the case where the vertices are positioned at $P_1 = (0,0)^T$, $P_2 = (2,0)^T$, and $P_3 = (1,1)^T$, as displayed in Fig. 1, the corresponding rigidity matrix is

$$M = \begin{array}{c} \\ e_{1,2} \\ e_{1,3} \\ e_{2,3} \end{array} \begin{array}{c} \begin{array}{cccccc} x_1 & y_1 & x_2 & y_2 & x_3 & y_3 \end{array} \\ \begin{pmatrix} -2 & 0 & 2 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 & -1 & 1 \end{pmatrix} \end{array}. \tag{3}$$

The example given above is straightforward since it is a planar linkage where all of the passive joints are revolute. If the revolute joints of this mechanism were replaced by spherical joints, such that the corresponding spatial mechanism was formed, the 3-dimensional graph would again consist of three vertices connected by three edges. The spatial manipulator examined in this paper consists of revolute and universal joints in addition to spherical joints, the corresponding sub-graphs for each of these joints are addressed here. A revolute joint in a spatial mechanism corresponds to two vertices which lie along the joint's axis. If the revolute joint is attached to a spherical joint, each vertex is connected to the vertex corresponding to the spherical joint in addition to each other [20]. A universal joint attached to the base corresponds to three adjacent vertices, two attached to the base, forming the base revolute axis, and the third, able to move with respect to the base, forming the moving revolute axis with one of the other two vertices. If the universal joint is connected to a revolute joint, then the vertices that form the moving axis are each connected to both of the vertices corresponding to the revolute joint.

A framework is rigid if, and only if, the row rank, herein referred to as rank, of its corresponding rigidity matrix is equal to $S(n, d)$. This is because as all infinitesimal motions must be in the null space of M, and $S(n, d)$ represents the size of the rigidity matrix without any trivial infinitesimal motions, it follows that if there exists any non-trivial motions within the null space of M, its rank must be less than $S(n, d)$. Therefore for a parallel robot, which is generally rigid, its corresponding rigidity matrix should be of full rank except for singular configurations in which it loses its inherent rigidity. In section V, computing the rank of the rigidity matrix is used as a steadfast method of determining whether or not a parallel robot has entered a singularity.

III. Kinematic Redundancy in Parallel Robots

A kinematically redundant parallel robot is a mechanism whose total number of degrees of freedom exceeds the number of degrees of freedom of the end-effector, and that is rigid when, and only when, all of its actuators are locked. Here we categorise such architectures into two different types: those which contain serial connected actuators and those which do not. Architectures with serially connected

Fig. 2. A family of kinematically redundant parallel robots with non-serially connected actuators proposed in the literature. The architectures, from left to right, were first presented in [21], [13], and [14], respectively.

actuators can be obtained by taking a non-redundant architecture and adding extra actuated joints to the existing limbs. Architectures which don't exhibit actuators connected in series contain at least two limbs that share a kinematic constraint between the base and the moving platform. Fig. 2 displays three instances of non-serially actuated kinematically redundant parallel architectures that have been proposed in the literature.

The architecture displayed on the left-hand side of Fig. 2, first presented in [21], is a planar mechanism that consists of four RPR legs, that is an actuated prismatic joint with a passive revolute joint at each of its ends, two of which join the end-effector to the base directly, and the the other two join the end-effector to a ternary link, which itself is connected to the base via a passive revolute joint. The second architecture, presented in [13], is also a planar mechanism that consists of four RPR legs. Two of the legs are connected to the base and the end-effector at separate points, the other two legs are joined to the end-effector at separate points and to a binary link at the same point, which in turn is connected to the base via a passive revolute joint. The third architecture, presented in [14], consists of a moving platform which is connected to the base by multiple redundant and non-redundant legs. A non-redundant leg consists of a prismatic joint which is joined to the platform via a spherical joint and to the base via a universal joint. A redundant leg consists of two prismatic actuators joined to the base at different points via universal joints, and to each other via a revolute joint, which in turn is connected to the platform via a spherical joint. The instance of the manipulator shown in Fig. 2 consists of three pairings of redundant and non-redundant legs, where the universal joints of each pairing are positioned upon of the same line. The six spherical joints connecting the legs to platform are located at three different positions; each position shares a joint between a redundant and non-redundant leg from different pairings.

The advantage that kinematically redundant architectures hold over their non-redundant counterparts is that it is possible to reconfigure the mechanism without changing the pose of the end-effector, meaning that the singularity locus in the robot's workspace is significantly reduced. Additionally, unlike redundantly actuated robots which are over constrained mechanisms, kinematically redundant architectures do not suffer from needless internal moments and forces being exerted onto the platform since the robot is only rigid when all of the actuators are locked. Furthermore, kinematically redundant architectures with non-serially connected actuators, such as those presented in Fig. 2, do not suffer from the accumulation of actuator errors along each of the limbs. In the following section, the method of calculating the Jacobian matrices, relating the robot's input joint velocity vector to the output velocity vector, is demonstrated for each of these three architectures.

IV. Calculation of the Jacobian

The relationship between the input joint velocities and the output velocity of the end-effector of a parallel robot can be described by the Jacobian matrices, $\mathbf{J}$ and $\mathbf{K}$, such that

$$\mathbf{J}\dot{\mathbf{c}} = \mathbf{K}\dot{\mathbf{q}} \tag{4}$$

where $\dot{\mathbf{c}}$ and $\dot{\mathbf{q}}$ denote the output and input velocity vectors, respectively. Traditionally, the robot is determined to be in a type-II singularity if $\mathbf{J}$ is singular. In this section, the method of calculating $\mathbf{J}$ for each of the three kinematically redundant parallel robots with non-serially connected actuators displayed in Fig. 2 is demonstrated. Unlike for non-redundant architectures, the method requires the elimination of at least one redundant output velocity variable in order to form the row(s) corresponding to the branches of the mechanism which join between the end-effector and the base; e.g. when two legs are joined to common link which, in turn, is joined to the base. This process of eliminating the redundant output variable(s) generates issues when performing singularity analysis, these problems are discussed in section V.

The methods of calculating the Jacobian for each of these three mechanisms is summarised below. The aim of this section is to highlight the need for the elimination of redundant joint velocities, the aim is not to give a detailed account of how the Jacobian is calculated from start to finish. For a more comprehensive detailing of each method, the reader is referred to the detailed Jacobian calculations in the multimedia material.

A. Architecture 1 - 1^{st} Planar Case

Firstly, let's consider the robot architecture displayed on the left-hand side of Fig. 2, the corresponding kinematic diagram of this architecture is shown in Fig. 3. The moving platform (P_6P_7) is connected directly to the base via two RPR legs at P_1 and P_2, namely legs 1 and 2, and to the ternary link via the two other RPR legs at P_4 and P_5, namely legs 3 and 4, which is connected to the base itself via a revolute joint at P_3. A fixed reference frame, Oxy, is attached to the base and a moving frame, $P_e x'y'$, is attached to the moving platform. The orientation of the platform, ϕ, is given by the angle taken anti-clockwise from the horizontal axis of the fixed frame to that of the moving frame, centred at $P_e(x, y)$. After forming the vector loop equations along each of the four legs, and taking the derivative with respect to time for each of them, we obtain

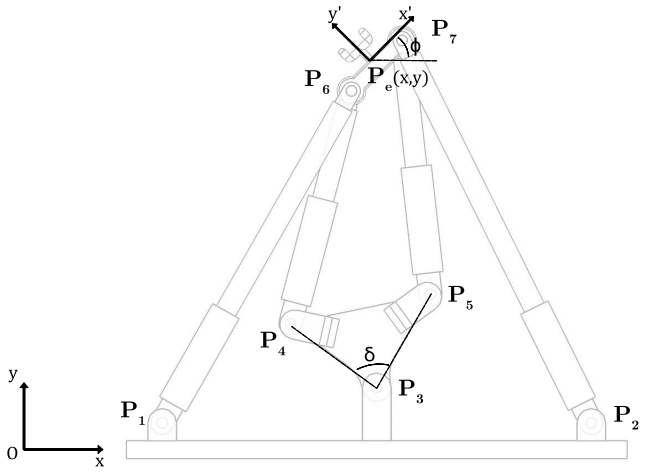

Fig. 3. The corresponding Kinematic diagram of the architecture displayed in the left-hand side of Fig. 2, that of the mechanism presented in [21].

$$\mathbf{p}_{1,6}^T(\dot{\mathbf{p}}_e - \dot{\mathbf{p}}_{6,e}) = \rho_1\dot{\rho}_1, \tag{5}$$

$$\mathbf{p}_{2,7}^T(\dot{\mathbf{p}}_e - \dot{\mathbf{p}}_{7,e}) = \rho_2\dot{\rho}_2, \tag{6}$$

$$\mathbf{p}_{4,6}^T(\dot{\mathbf{p}}_e - \dot{\mathbf{p}}_{3,4} - \dot{\mathbf{p}}_{6,e}) = \rho_3\dot{\rho}_3, \tag{7}$$

$$\mathbf{p}_{5,7}^T(\dot{\mathbf{p}}_e - \dot{\mathbf{p}}_{3,5} - \dot{\mathbf{p}}_{7,e}) = \rho_4\dot{\rho}_4. \tag{8}$$

where $\mathbf{p}_{i,j}$ denotes the vector from P_i to P_j, $\mathbf{p}_e$ denotes the position vector of P_e, ρ_i denotes the length of the prismatic actuators of the i^{th} leg of the manipulator, and dot notation is used to indicate a derivative with respect to time.

Since the output of the robot is the 3-dimensional velocity vector, $\dot{\mathbf{c}} = (\dot{x}, \dot{y}, \dot{\phi})^T$, and the input is the 4-dimensional velocity vector, $\dot{\mathbf{q}} = (\dot{\rho}_1, \dot{\rho}_2, \dot{\rho}_3, \dot{\rho}_4)^T$, the Jacobian matrices, $\mathbf{J}$ and $\mathbf{K}$, are of dimension 3×3 and 3×4, respectively. The first two rows are formed by equations (5) and (6), whereas the third row is formed by combining equations (7) and (8) through the elimination of $\dot{\mathbf{p}}_{3,5}$; the vector which corresponds to the redundant output variable.

Since P_3, P_4, and P_5 are all connected to the same ternary link, the following relation exists:

$$\mathbf{p}_{3,4} = \frac{d_{3,4}}{d_{3,5}} \begin{bmatrix} cos(\delta) & -sin(\delta) \\ sin(\delta) & cos(\delta) \end{bmatrix} \mathbf{p}_{3,5} = \lambda\mathbf{M}\mathbf{p}_{3,5} \tag{9}$$

where δ is the angle taken anti-clockwise from $\mathbf{p}_{3,5}$ to $\mathbf{p}_{3,4}$. Using this relation, equations (7) and (8) can be combined to produce the following equation

$$\begin{bmatrix} \mathbf{p}_{4,6}^T(\dot{\mathbf{p}}_e - \dot{\mathbf{p}}_{6,e}) - \rho_3\dot{\rho}_3 \\ \mathbf{p}_{5,7}^T(\dot{\mathbf{p}}_e - \dot{\mathbf{p}}_{7,e}) - \rho_4\dot{\rho}_4 \end{bmatrix} = \begin{bmatrix} \mathbf{p}_{4,6}^T\lambda\mathbf{M} \\ \mathbf{p}_{5,7}^T \end{bmatrix} \dot{\mathbf{p}}_{3,5}, \tag{10}$$

and then $\dot{\mathbf{p}}_{3,5}$ can be made the subject by

$$\dot{\mathbf{p}}_{3,5} = \mathbf{N} \begin{bmatrix} \mathbf{p}_{4,6}^T(\dot{\mathbf{p}}_e - \dot{\mathbf{p}}_{6,e}) - \rho_3\dot{\rho}_3 \\ \mathbf{p}_{5,7}^T(\dot{\mathbf{p}}_e - \dot{\mathbf{p}}_{7,e}) - \rho_4\dot{\rho}_4 \end{bmatrix}, \tag{11}$$

where

$$\mathbf{N} = \begin{bmatrix} \mathbf{p}_{4,6}^T\lambda\mathbf{M} \\ \mathbf{p}_{5,7}^T \end{bmatrix}^{-1}.$$

Since the distance between P_3 and P_5 is constant,

$$\mathbf{p}_{3,5}^T\dot{\mathbf{p}}_{3,5} = 0. \tag{12}$$

The redundant output variable, $\dot{\mathbf{p}}_{3,5}$, is then eliminated by substituting (11) into (12). By expanding the velocity vectors, we obtain

$$\mathbf{p}_{3,5}^T\mathbf{N} \begin{bmatrix} \mathbf{p}_{4,6}^T \\ \mathbf{p}_{5,7}^T \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} - \mathbf{p}_{3,5}^T\mathbf{N} \begin{bmatrix} \mathbf{p}_{4,6}^T\mathbf{E}\mathbf{p}_{6,e} \\ \mathbf{p}_{5,7}^T\mathbf{E}\mathbf{p}_{7,e} \end{bmatrix} \dot{\phi} = \mathbf{p}_{3,5}^T\mathbf{N} \begin{bmatrix} \rho_3\dot{\rho}_3 \\ \rho_4\dot{\rho}_4 \end{bmatrix}, \tag{13}$$

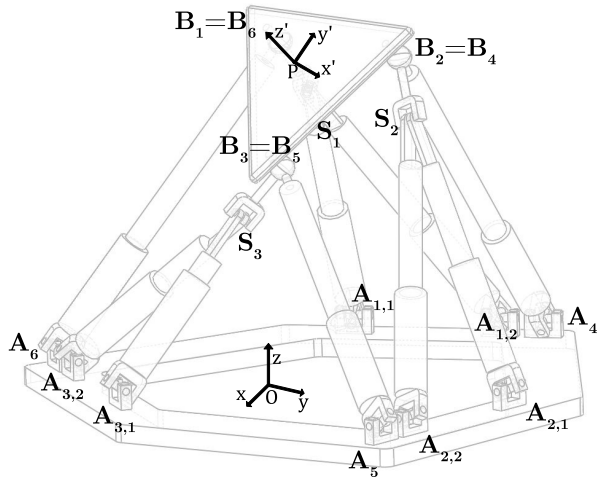

Fig. 4. The corresponding kinematic diagram of the architecture displayed in the right-hand side of Fig. 2, that of the mechanism presented in [14].

where

$$\mathbf{E} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}.$$

By similarly expanding the velocity vectors of equations (5) and (6), the Jacobian matrices $\mathbf{J}$ and $\mathbf{K}$ can be formed, such that $\mathbf{J}$ is given by

$$\mathbf{J} = \begin{bmatrix} \mathbf{p}_{1,6}^T & -\mathbf{p}_{1,6}^T\mathbf{E}\mathbf{p}_{6,e} \\ \mathbf{p}_{2,7}^T & -\mathbf{p}_{2,7}^T\mathbf{E}\mathbf{p}_{7,e} \\ \mathbf{p}_{3,5}^T\mathbf{N} \begin{bmatrix} \mathbf{p}_{4,6}^T \\ \mathbf{p}_{5,7}^T \end{bmatrix} & -\mathbf{p}_{3,5}^T\mathbf{N} \begin{bmatrix} \mathbf{p}_{4,6}^T\mathbf{E}\mathbf{p}_{6,e} \\ \mathbf{p}_{5,7}^T\mathbf{E}\mathbf{p}_{7,e} \end{bmatrix} \end{bmatrix}. \tag{14}$$

B. Architecture 2 - *Spatial Case*

Here the Jacobian is calculated for the kinematically redundant spatial manipulator displayed in the right-hand side of Fig. 2, the corresponding kinematic diagram of which is displayed in Fig. 4. As mentioned above, the manipulator consists of a moving platform attached to the base by three redundant and three non-redundant legs. The non-redundant legs consist of an actuated prismatic joint which is connect to the base via a universal joint, at point A_i, and to the moving platform via a spherical joint, at point B_i, where $i = 4, 5, 6$. The redundant legs consist of two actuated prismatic joints joined to the base at points $A_{i,1}$ and $A_{i,2}$, to each other via a revolute joint at S_i, and to the moving platform via a spherical joint at B_i, where $i = 1, 2, 3$. The six spherical joints attached to the platform are positioned in coincident pairs. A fixed reference frame $Oxyz$ is attached to the base and a moving reference frame $Px'y'z'$ is attached to the moving platform. In [14], the Jacobian is calculated for a manipulator with an unspecified number of redundant legs, here the same method is simplified for a manipulator with three redundant and three non-redundant legs. The position vectors of the universal joints on the base, $A_{i,j}$ and A_i for the redundant and non-redundant legs respectively, are denoted by $\mathbf{a}_{i,j}$ and $\mathbf{a}_i$. The position vectors for the spherical joints, B_i, on the platform are given by $\mathbf{b}_i$, and the position vectors of each revolute joint, S_i, are denoted by $\mathbf{s}_i$. The Jacobian, $\mathbf{J}$, of this robot is a 6×6 matrix, where three of the rows correspond to the redundant legs and three correspond to the non-redundant legs. Here, the steps required to compute the rows corresponding the redundant legs are shown as this is where the elimination of the redundant variables occurs.

The position of the i^{th} platform joint in terms of $\mathbf{Q}$, the matrix denoting the orientation of the platform, and $\mathbf{v}_{i,0}$, the position of the joint in the moving frame,is given by

$$\mathbf{b}_i = \mathbf{p} + \mathbf{Q}\mathbf{v}_{i,0}, i = 1, ..., 6. \tag{15}$$

For the i^{th} redundant leg, the following constraint equations are written

$$(\mathbf{s}_i - \mathbf{a}_{i,j})^T(\mathbf{s}_i - \mathbf{a}_{i,j}) = \rho_{i,j}^2, \tag{16}$$

$$(\mathbf{s}_i - \mathbf{b}_i)^T(\mathbf{s}_i - \mathbf{b}_i) = l_i^2, \tag{17}$$

where l_i denotes the length of the link which joins S_i and B_i, and $j = 1, 2$. Given that the joints $A_{i,1}$, $A_{i,2}$, S_i, and B_i are coplanar, if we define a unit vector $\mathbf{e}_i$ which passes through the base joints of redundant leg i, the following relationship must hold

$$[(\mathbf{b}_i - \mathbf{a}_{i,1}) \times \mathbf{e}_i]^T(\mathbf{s}_i - \mathbf{a}_{i,1}) = 0. \tag{18}$$

By differentiating equations (16) and (18), the following is obtained

$$\begin{bmatrix} (\mathbf{s}_i - \mathbf{a}_{i,1})^T \\ (\mathbf{s}_i - \mathbf{a}_{i,2})^T \\ [(\mathbf{b}_i - \mathbf{a}_{i,1}) \times \mathbf{e}_i]^T \end{bmatrix} \dot{\mathbf{s}}_i = \mathbf{H}_i\dot{\mathbf{s}}_i = \begin{bmatrix} \rho_{i,1}\dot{\rho}_{i,1} \\ \rho_{i,2}\dot{\rho}_{i,2} \\ [(\mathbf{s}_i - \mathbf{a}_{i,1}) \times \mathbf{e}_i]^T\dot{\mathbf{b}}_i \end{bmatrix}. \tag{19}$$

Equation (19) is solved for $\dot{\mathbf{s}}_i$ by taking the matrix inverse of $\mathbf{H}_i$, such that

$$\dot{\mathbf{s}}_i = \mathbf{H}_i^{-1} \begin{bmatrix} \rho_{i,1}\dot{\rho}_{i,1} \\ \rho_{i,2}\dot{\rho}_{i,2} \\ [(\mathbf{s}_i - \mathbf{a}_{i,1}) \times \mathbf{e}_i]^T\dot{\mathbf{b}}_i \end{bmatrix}, \tag{20}$$

where $\mathbf{H}_i^{-1}$ can be expressed as

$$\mathbf{H}_i^{-1} = \frac{Adj(\mathbf{H}_i)}{det(\mathbf{H}_i)}, \tag{21}$$

$Adj(\mathbf{H}_i)$ is the adjoint of matrix $\mathbf{H}_i$ and $det(\mathbf{H}_i)$ is the determinant, which herein will be denoted by μ_i. These can be expressed algebraically by

$$det(\mathbf{H}_i) = \mu_i = [(\mathbf{s}_i - \mathbf{a}_{i,1}) \times (\mathbf{s}_i - \mathbf{a}_{i,2})]^T[(\mathbf{b}_i - \mathbf{a}_{i,1}) \times \mathbf{e}_i] \tag{22}$$

and

$$Adj(\mathbf{H}_i) = \begin{bmatrix} \mathbf{h}_{i,1} & \mathbf{h}_{i,2} & \mathbf{h}_{i,3} \end{bmatrix} \tag{23}$$

where

$$\mathbf{h}_{i,1} = (\mathbf{s}_i - \mathbf{a}_{i,2}) \times [(\mathbf{b}_i - \mathbf{a}_{i,1}) \times \mathbf{e}_i], \tag{24}$$

$$\mathbf{h}_{i,2} = [(\mathbf{b}_i - \mathbf{a}_{i,1}) \times \mathbf{e}_i] \times (\mathbf{s}_i - \mathbf{a}_{i,1}), \tag{25}$$

$$\mathbf{h}_{i,3} = (\mathbf{s}_i - \mathbf{a}_{i,1}) \times (\mathbf{s}_i - \mathbf{a}_{i,2}). \tag{26}$$

Now equation (20) can be rewritten as

$$\dot{\mathbf{s}}_i = \frac{1}{\mu_i}(\mathbf{h}_{i,1}\rho_{i,1}\dot{\rho}_{i,1} + \mathbf{h}_{i,2}\rho_{i,2}\dot{\rho}_{i,2} + \mathbf{h}_{i,3}[(\mathbf{s}_i - \mathbf{a}_{i,1}) \times \mathbf{e}_i]^T\dot{\mathbf{b}}_i). \tag{27}$$

By taking the derivative of (17), one obtains

$$(\mathbf{s}_i - \mathbf{b}_i)^T\dot{\mathbf{s}}_i = (\mathbf{s}_i - \mathbf{b}_i)^T\dot{\mathbf{b}}_i, \tag{28}$$

and substituting (27) and the derivative of (15) into (28) gives

$$(\mathbf{s}_i - \mathbf{b}_i)^T\dot{\mathbf{p}} + [\mathbf{Q}\mathbf{v}_{i,0} \times (\mathbf{s}_i - \mathbf{b}_i)]^T\boldsymbol{\omega} = (\mathbf{s}_i - \mathbf{b}_i)^T\mathbf{m}_i\dot{\rho}_{i,1} + (\mathbf{s}_i - \mathbf{b}_i)^T\mathbf{n}_i\dot{\rho}_{i,2} \tag{29}$$

where

$$\mathbf{m}_i = \frac{\rho_{i,1}}{\mu_i}[(\mathbf{s}_i - \mathbf{a}_{i,2}) \times [(\mathbf{b}_i - \mathbf{a}_{i,1}) \times \mathbf{e}_i]], \tag{30}$$

$$\mathbf{n}_i = \frac{\rho_{i,2}}{\mu_i}[[(\mathbf{b}_i - \mathbf{a}_{i,1}) \times \mathbf{e}_i] \times (\mathbf{s}_i - \mathbf{a}_{i,1})]. \tag{31}$$

The output velocity vector of the manipulator is given by $\dot{\mathbf{c}} = (\dot{\mathbf{p}}^T, \boldsymbol{\omega}^T)^T$ and the vector of actuated joint velocities is given by

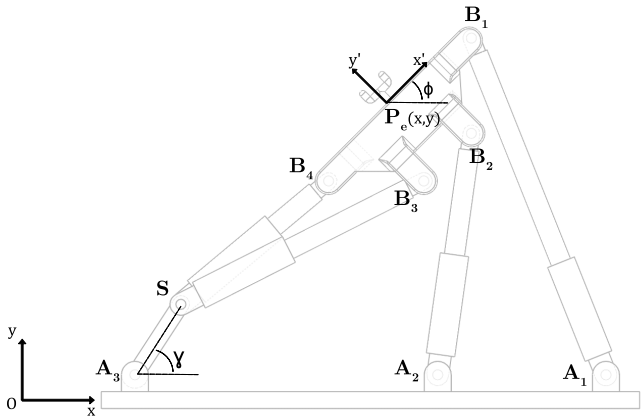

Fig. 5. The corresponding Kinematic diagram of the architecture displayed in the centre of Fig. 2, that of the mechanism presented in [13].

$\dot{\mathbf{q}} = (\dot{\rho}_{1,1}, \dot{\rho}_{1,2}, \dot{\rho}_{2,1}, \dot{\rho}_{2,2}, \dot{\rho}_{3,1}, \dot{\rho}_{3,2}, \dot{\rho}_4, \dot{\rho}_5, \dot{\rho}_6)^T$. Equation (29) is used to construct the first three rows of the Jacobian which correspond to the redundant legs of the manipulator. Then, along with the latter three rows which correspond to the non-redundant legs, the Jacobian matrices $\mathbf{J}$ and $\mathbf{K}$ can be computed. Matrix $\mathbf{J}$ is given by

$$\mathbf{J} = \begin{bmatrix} (\mathbf{s}_1 - \mathbf{b}_1)^T & [\mathbf{Q}\mathbf{v}_{1,0} \times (\mathbf{s}_1 - \mathbf{b}_1)^T] \\ (\mathbf{s}_2 - \mathbf{b}_2)^T & [\mathbf{Q}\mathbf{v}_{2,0} \times (\mathbf{s}_2 - \mathbf{b}_2)^T] \\ (\mathbf{s}_3 - \mathbf{b}_3)^T & [\mathbf{Q}\mathbf{v}_{3,0} \times (\mathbf{s}_3 - \mathbf{b}_3)^T] \\ (\mathbf{b}_4 - \mathbf{a}_4)^T & [\mathbf{Q}\mathbf{v}_{4,0} \times (\mathbf{b}_4 - \mathbf{a}_4)^T] \\ (\mathbf{b}_5 - \mathbf{a}_5)^T & [\mathbf{Q}\mathbf{v}_{5,0} \times (\mathbf{b}_5 - \mathbf{a}_5)^T] \\ (\mathbf{b}_6 - \mathbf{a}_6)^T & [\mathbf{Q}\mathbf{v}_{6,0} \times (\mathbf{b}_6 - \mathbf{a}_6)^T] \end{bmatrix}. \tag{32}$$

C. Architecture 3 - 2^{nd} Planar Case

The final mechanism under inspection, displayed in the centre of Fig. 2, is the kinematically redundant planar parallel architecture presented in [13], the kinematic diagram of which is shown in Fig. 5. The architecture consists of four RPR legs, two which of are joined to the base at points A_1 and A_2, and to the moving platform at B_1 and B_2. The other two are joined to the platform at points B_3 and B_4, and are joined to an additional link at the same point, S, which in turn is connected to the base via a revolute joint centred at A_3. A fixed reference frame Oxy is attached to the base, and a moving reference frame $Px'y'$ is attached to the platform at point $P(x, y)$; the orientation of the platform, ϕ, is defined by the angle taken anti-clockwise from the horizontal axis of the fixed reference frame to that of the moving reference frame. The position vectors of points A_i, B_i, S, and P are denoted by $\mathbf{a}_i$, $\mathbf{b}_i$, $\mathbf{s}$, and $\mathbf{p}$ respectively. The Cartesian coordinates of the platform are given by $\mathbf{c} = (x, y, \phi)^T$. The distance between joints A_i and B_i, $i = 1, 2$, and S and B_i, $i = 3, 4$, is denoted by ρ_i, corresponding to the lengths of the prismatic actuators. The orientation of link A_3S relative to the fixed reference frame is given by γ. Firstly, the constraint equations in terms of the square of the length of each prismatic actuator, ρ_i^2, and the square of the length of link A_3S, l_i^2, are formed, and their derivatives are obtained as

$$(\mathbf{b}_i - \mathbf{a}_i)^T(\dot{\mathbf{p}} + \dot{\phi}\mathbf{E}\boldsymbol{\nu}_i) = \rho_i\dot{\rho}_i, i = 1, 2, \tag{33}$$

$$(\mathbf{b}_i - \mathbf{s})^T(\dot{\mathbf{p}} + \dot{\phi}\mathbf{E}\boldsymbol{\nu}_i - \dot{\mathbf{s}}) = \rho_i\dot{\rho}_i, i = 3, 4 \tag{34}$$

$$(\mathbf{s} - \mathbf{a}_3)^T\dot{\mathbf{s}} = 0 \tag{35}$$

where $\boldsymbol{\nu}_i = \mathbf{Q}\boldsymbol{\nu}_{0,i}$. Equations (34) are then combined to form the matrix equation

$$\mathbf{G}\dot{\mathbf{c}} - \mathbf{h} = \mathbf{H}\dot{\mathbf{s}} \tag{36}$$

such that

$$\begin{bmatrix} \mathbf{f}^T & \mathbf{f}^T\mathbf{E}\boldsymbol{\nu}_3 \\ \mathbf{m}^T & \mathbf{m}^T\mathbf{E}\boldsymbol{\nu}_4 \end{bmatrix} \dot{\mathbf{c}} - \begin{bmatrix} \rho_3\dot{\rho}_3 \\ \rho_4\dot{\rho}_4 \end{bmatrix} = \begin{bmatrix} \mathbf{f}^T \\ \mathbf{m}^T \end{bmatrix} \dot{\mathbf{s}}$$

where $\mathbf{f}^T = (\mathbf{b}_3 - \mathbf{s})^T$ and $\mathbf{m}^T = (\mathbf{b}_4 - \mathbf{s})^T$. Equation (36) is then rearranged to make $\dot{\mathbf{s}}$ the subject by taking the inverse of matrix $\mathbf{H}$, such that

$$\dot{\mathbf{s}} = \mathbf{N}(\mathbf{G}\dot{\mathbf{c}} - \mathbf{h}) \tag{37}$$

where

$$\mathbf{N} = \mathbf{H}^{-1} = \frac{1}{\mathbf{f}^T\mathbf{E}\mathbf{m}} \begin{bmatrix} \mathbf{E}\mathbf{m} & -\mathbf{E}\mathbf{f} \end{bmatrix}.$$

The redundant variable, $\dot{\mathbf{s}}$, is then eliminated by substituting (37) into (35), such that

$$(\mathbf{s} - \mathbf{a}_3)^T\mathbf{N}\mathbf{G}\dot{\mathbf{c}} = (\mathbf{s} - \mathbf{a}_3)^T \begin{bmatrix} \frac{\mathbf{E}\mathbf{m}\rho_3}{\mathbf{f}^T\mathbf{E}\mathbf{m}} & \frac{-\mathbf{E}\mathbf{f}\rho_4}{\mathbf{f}^T\mathbf{E}\mathbf{m}} \end{bmatrix} \begin{bmatrix} \dot{\rho}_3 \\ \dot{\rho}_4 \end{bmatrix} \tag{38}$$

where

$$\mathbf{N}\mathbf{G} = \frac{\mathbf{E}}{\mathbf{f}^T\mathbf{E}\mathbf{m}} \begin{bmatrix} (\mathbf{m}\mathbf{f}^T - \mathbf{f}\mathbf{m}^T) & (\mathbf{m}\mathbf{f}^T\mathbf{E}\boldsymbol{\nu}_3 - \mathbf{f}\mathbf{m}^T\mathbf{E}\boldsymbol{\nu}_4) \end{bmatrix}. \tag{39}$$

This may be further simplified to

$$\mathbf{N}\mathbf{G} = \begin{bmatrix} \mathbf{1} & \mathbf{E}(\mathbf{s} - \mathbf{p}) \end{bmatrix} \tag{40}$$

where $\mathbf{1}$ denotes the 2×2 identity matrix. The details of this simplification are given in the detailed Jacobian calculations in the multimedia material, however this is not the focus; if either (39) or (40) are used to form the Jacobian, the same problems still manifest themselves. These issues are generated through performing the matrix inverse, $\mathbf{H}^{-1}$, and the elimination of the redundant variable, $\dot{\mathbf{s}}$. The first two rows of the Jacobian Matrices, $\mathbf{J}$ and $\mathbf{K}$, can then be formed from (33), and the third row can be obtained by substituting (40) into (38), such that matrix $\mathbf{J}$ is given by

$$\mathbf{J} = \begin{bmatrix} (\mathbf{b}_1 - \mathbf{a}_1)^T & (\mathbf{b}_1 - \mathbf{a}_1)^T\mathbf{E}\boldsymbol{\nu}_1 \\ (\mathbf{b}_2 - \mathbf{a}_2)^T & (\mathbf{b}_2 - \mathbf{a}_2)^T\mathbf{E}\boldsymbol{\nu}_2 \\ (\mathbf{s} - \mathbf{a}_3)^T & (\mathbf{s} - \mathbf{a}_3)^T\mathbf{E}(\mathbf{s} - \mathbf{p}) \end{bmatrix}. \tag{41}$$

V. Limitations of the Jacobian

In this section, some examples of the problems with using Jacobian-based methods of singularity analysis for kinematically redundant robots with non-serially connected actuators are demonstrated. The Jacobian is computed for each of the three architectures presented in Fig. 2 whilst in configurations where these problems manifest themselves. The singularity analysis is conducted using the inverse of the 2-norm condition number of the Jacobian, and the results are assessed by constructing the rigidity matrix and calculating its rank. The rigidity matrices for each of the three mechanisms are provided in the multimedia material.

A. 1^{st} Planar Case - False Positive of the Jacobian

In this first example, we consider a mechanism which has the same architecture as that presented in the left-hand side of Fig. 2. Let's consider the configuration of this mechanism where $\mathbf{p}_1 = (0, 0)^T$, $\mathbf{p}_2 = (3, 0)^T$, $\mathbf{p}_3 = (2.5, 1)^T$, $\mathbf{p}_4 = (1.79, 1.71)^T$, $\mathbf{p}_5 = (2.5, 2)^T$, $\mathbf{p}_6 = (1.41, 2.63)^T$, and $\mathbf{p}_7 = (2.88, 2.92)^T$; the corresponding kinematic diagram is shown in Fig. 6. The Jacobian, $\mathbf{J}$, is obtained by inputting these values into (14). The inverse of the 2-norm condition number of the obtained Jacobian is zero, suggesting that the robot is in a singularity. However, the rigidity matrix of the mechanism in this configuration has full rank, this indicates that the robot is not in a singularity, despite the fact that the inverse of the condition number of the Jacobian suggests the robot is in a singularity.

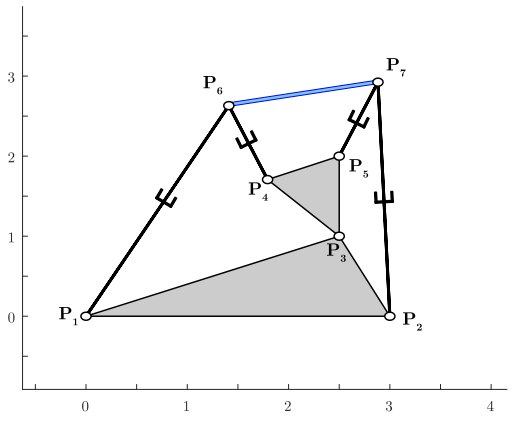

Fig. 6. Configuration of the robot proposed in [21] where the inverse of the condition number of the Jacobian is zero, suggesting that the configuration is singular, but the rigidity matrix is of full rank, indicating that the mechanism is not in a singularity.

B. Spatial Case - False Negative of the Jacobian

Now we turn our attention to the spatial manipulator. In this case, the issue is that it is possible to have a configuration where the robot enters a singularity, indicated by the fact that the rigidity matrix loses rank, but the determinant of the Jacobian is non-zero. In [14], where the architecture is presented and the Jacobian is calculated, the authors state that an assumption of this mechanism is that the legs never lie in the base plane. However, since the inverse of the condition of the Jacobian does not approach zero as the robot nears such a configuration, it does not act as a reliable method of analysing how the performance of the robot deteriorates near all singularities, making it a bad basis for path planning algorithms. In the following example, the robot is initially in a non-singular pose and the platform follows a trajectory towards the configuration in which the revolute joint of one of the redundant legs, $\mathbf{s}_3$, lies on the line passing through the base joints $\mathbf{a}_{3,1}$ and $\mathbf{a}_{3,2}$. A fixed reference frame is attached to the base and the base joints are positioned at: $\mathbf{a}_{1,1} = (-1.4, 0, 0)^T$, $\mathbf{a}_{1,2} = (-1.0, -0.69, 0)^T$, $\mathbf{a}_{2,1} = (0.7, 1.21, 0)^T$, $\mathbf{a}_{2,2} = (-0, 1, 1.21, 0)^T$, $\mathbf{a}_{3,1} = (0.7, -1.21, 0)^T$, $\mathbf{a}_{3,2} = (1.1, -0.52, 0)^T$, $\mathbf{a}_4 = (-1.5, 0.17, 0)^T$, $\mathbf{a}_5 = (0.9, 1.21, 0)^T$, and $\mathbf{a}_6 = (0.6, -1.39, 0)^T$. When in the initial pose, the platform joints are positioned at: $\mathbf{b}_1 = (-1.18, -0.43, 1.68)^T$, $\mathbf{b}_2 = (-1.18, 0.43, 1.68)^T$, and $\mathbf{b}_3 = (0.35, 0, 1.15)^T$, and the revolute joints on the redundant legs are positioned at: $\mathbf{s}_1 = (-0.23, -0.43, 1.59)^T$, $\mathbf{s}_2 = (-0.15, 0.47, 1.59)^T$, and $\mathbf{s}_3 = (0.39, -0.06, 1.07)^T$; all coordinates are given to two decimal places.

The platform follows a linear trajectory such that the final pose of the platform is given by $\mathbf{b}_1 = (0.33, -1.24, 0.60)^T$, $\mathbf{b}_2 = (0.33, -0.38, 0.60)^T$, and $\mathbf{b}_3 = (0.86, -0.81, 0.07)^T$. The revolute joints of the redundant links are positioned throughout the trajectory such that the line passing through the link $\mathbf{b}_i\mathbf{s}_i$ passes through the midpoint of base joints $\mathbf{a}_{i,1}$ and $\mathbf{a}_{i,2}$; their positions at the end of the trajectory are $\mathbf{s}_1 = (0.25, -1.19, 0.57)^T$, $\mathbf{s}_2 = (0.33, -0.28, 0.57)^T$, and $\mathbf{s}_3 = (0.90, -0.87, 0)^T$. Fig. 7 shows the initial and final pose of the manipulator during this trajectory.

The trajectory is discretised into 101 steps, and the value of $1/\kappa(J)$ (the inverse of the 2-norm condition number of J), is displayed at each step in Fig. 8. The value of $1/\kappa(J)$ does not go to zero as the manipulator reaches the final pose of the trajectory, where the revolute joint $\mathbf{s}_3$ lies directly between $\mathbf{a}_{3,1}$ and $\mathbf{a}_{3,2}$. However, in this configuration the manipulator is in a type-II singularity, this is

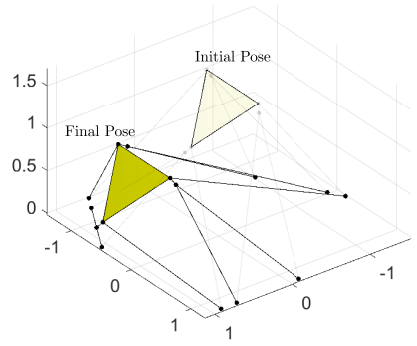

Fig. 7. The initial and final configurations of the example kinematically redundant spatial manipulator as it moves from a non-singular pose into a singularity.

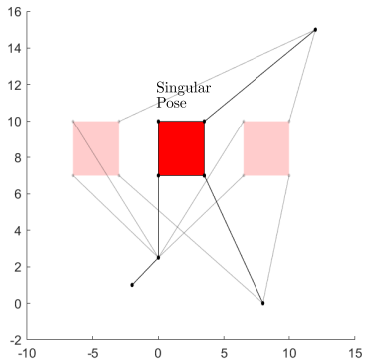

Fig. 9. Trajectory of the example kinematically redundant planar parallel robot, passing through a type-II singularity for which the Jacobian stays non-singular.

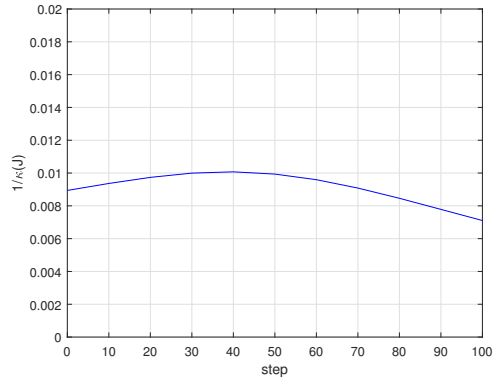

Fig. 8. Inverse of the condition number of the Jacobian, $1/\kappa(J)$, for spatial manipulator at each step of the trajectory, showing that the Jacobian does not approach becoming singular as the robot moves into a type-II singularity.

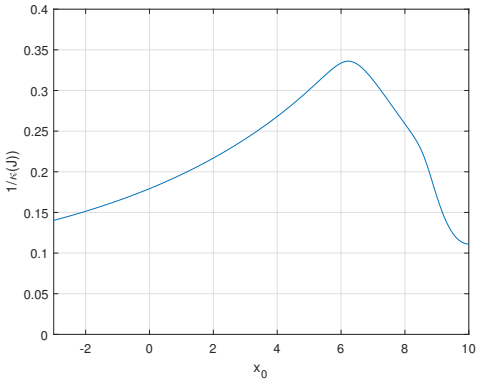

Fig. 10. Value of $1/\kappa(J)$ as the planar parallel robot passes through the singularity; the configuration where the limbs joined to the platform and the redundant link become collinear, occurring at $x_0 = 3.5$.

determined by formulating the rigidity matrix of the mechanism and computing its rank at each step. The rank of the rigidity matrix at each step is 88, except for the last step at which it drops to 87, indicating that the mechanism has entered a type-II singularity at this point and lost its rigidity.

C. 2^{nd} Planar Case - False Negative of the Jacobian

In this final example, the 2^{nd} planar architecture is examined as it moves from an initial non-singular configuration through a singularity. The trajectory is depicted in Fig. 9, the transparent instances show the initial and final non-singular configurations of the mechanism whereas the opaque instance shows the singular configuration. The base joints are positioned at: $\mathbf{a}_1 = (12, 15)^T$, $\mathbf{a}_2 = (8, 0)^T$, and $\mathbf{a}_3 = (-2, 1)^T$, and joint S is always positioned at $\mathbf{s} = (0, 2.5)^T$ throughout the trajectory. In the initial configuration, the platform joints are positioned at $\mathbf{b}_1 = (-3, 10)^T$, $\mathbf{b}_2 = (-3, 7)^T$, $\mathbf{b}_3 = (-6.5, 7)^T$, and $\mathbf{b}_4 = (-6.5, 10)^T$. The platform then moves along a horizontal trajectory to the right, passing through a configuration where the links $SB3$ and $SB4$ become collinear. The value of the inverse of the 2-norm condition number, $1/\kappa(J)$, is plotted against the x coordinate of platform joint $\mathbf{b}_1$ in Fig. 10. This case is similar to that reported in example 2, in that the value of $1/\kappa(J)$ does not approach zero as the robot nears a singularity, when $x_0 = 3.5$.

VI. Discussion

It is clear that these shortcomings of the Jacobian as a means of singularity detection have serious implications in terms of path planning algorithms for kinematically redundant parallel robots with non-serially connected actuators. For cases similar to the first example, the feasible workspace of the mechanism would needlessly be restricted since the Jacobian becomes singular in configurations where the robot is not in a singularity. Whereas for cases similar to the second two examples, any path planning algorithms based on the Jacobian would run the risk of moving the robot into a configuration where its performance may deteriorate significantly.

The issues are generated by the need to eliminate a, or multiple, redundant variables. For the first example, the determinant of the Jacobian becomes singular because the matrix that is inverted to obtain $\mathbf{N}$, in equation (11), itself becomes singular due to a linear dependence between its rows. Although this accounts for the singularity from a mathematical perspective, it does not translate into a physical meaning for the singularity. The geometric conditions for this instance to occur are that if the line which passes through leg 3 of the manipulator is rotated by δ in the clockwise direction, and the result is a collinearity with the line which passes through leg 4. In addition to computing the rank of the corresponding rigidity matrix, it is also possible to verify that this configuration is non-singular by performing the singularity analysis via instantaneous centres of rotation [10], [22], [23]. This analysis is not conducted here for the

sake of brevity, however the reader is referred to [10] for the method of performing the analysis on the same architecture as that in example 1, but with RRR instead of RPR legs. The result is that, when in the configuration detailed in example 1, the instantaneous centres of rotation between the platform and the base, for each of the four equivalent mechanisms where all but one of the actuators are locked, are all determinable and do not coincide with one another, indicating that the robot is not in a singularity.

The failure of the Jacobian in examples 2 and 3 is different to example 1; the inverse of the condition number of the Jacobian is non-zero, but we know that the robot is in a singularity as the corresponding rigidity matrix is rank deficient. Although the architectures in examples 2 and 3 correspond to spatial and planar cases respectively, the reasons which cause the Jacobian to fail in both examples are similar and so we will treat them both here. In example 2, the redundant variable $\mathbf{s}_i$ is eliminated by taking the inverse of matrix $\mathbf{H}_i$ so that equation (20) can be substituted into (28). In order to perform this matrix inverse, the determinant of $\mathbf{H}_i$, denoted μ_i, is taken; which equals zero when the prismatic actuators along $A_{i,1}S$ and $A_{i,2}S$ are collinear. It can be seen that (29) is obtained by substituting (27), that which contains μ_i, into (28). However, since the third term of the right-hand side of (27) is orthogonal to $(\mathbf{s}_i - \mathbf{b}_i)$, the product of them equals zero and hence μ_i does not feature in the Jacobian matrix, $\mathbf{J}$. Similarly in example 3, the term $\mathbf{NG}$ in (39) is simplified to (40), such that the coefficient $1/\mathbf{f}^T\mathbf{Em}$ is cancelled. An alternative is to not perform these simplifications such that the determinants of of matrices $\mathbf{H}_i$ and $\mathbf{H}$, respectively, remain present in the Jacobian; however this means that although $det(J) = 0$ when the robot is in this configuration, it does not smoothly approach zero as the robot approaches the configuration; therefore unless, the robot is positioned precisely in such a pose, the singularity will not be detected. It is also possible to detect these singularities by generating the so-called 'extended Jacobian', by including the time derivative of the redundant variable in the cartesian velocity vector [13], [24]. However, the use of this technique for path planning algorithms is limited as the time derivative of the redundant variable must also be selected in order to solve the inverse kinematics.

We summarise the findings presented above by advising developers of kinematically redundant parallel robots with non-serially connected actuators to use non-Jacobian based methods of path planning and singularity analysis. Computing the rank of rigidity matrix of the underlying graph of the mechanism is a reliable method of determining whether or not a particular configuration is singular or not, but does not help with informing how close the robot is to a singularity. It is clear that other methods need to be developed which do not exhibit the aforementioned issues detailed above.

VII. Conclusion

This paper details the shortcomings of using Jacobian-based methods of singularity analysis on kinematically redundant parlallel robots which exhibit no actuators connected in series. Three example mechanisms are examined in particular configurations where the Jacobian fails as a means of singularity analysis, in one case the determinant of the Jacobian equals zero when the robot is not in a singularity (false positive), in the other cases it fails to go to zero as the robot approaches a singularity (false negative). The robot is determined herein to be in a singularity by computing the rank of the rigidity matrix of the underlying graph of the mechanism; this matrix becomes rank deficient if the mechanism enters a type-II singularity and is full rank when in a non-singular configuration. The problems with the Jacobian for these types of parallel robots arise due to the need to eliminate a redundant variable(s). We summarise the paper by instructing developers of parallel robots with similar architectures to use non-Jacobian based methods of singularity detection or path planning algorithms. Indeed, future work may include the development of geometric methods for such purposes.

References

[1] J. Borràs, F. Thomas, and C. Torras, "On δ-transforms," *IEEE Transactions on Robotics*, vol. 25, no. 6, pp. 1225–1236, Dec 2009.

[2] C. Gosselin and J. Angeles, "Singularity Analysis of Closed-Loop Kinematic Chains," *IEEE Transactions on Robotics*, vol. 6, pp. 281–290, Jun. 1990.

[3] D. Zlatanov, R. G. Fenton, and B. Benhabib, "Singularity analysis of mechanisms and robots via a motion-space model of the instantaneous kinematics," in *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*. IEEE, 1994, pp. 980–985.

[4] D. Zlatanov, I. A. Bonev, and C. M. Gosselin, "Constraint singularities of parallel mechanisms," in *Proceedings 2002 IEEE International Conference on Robotics and Automation*, vol. 1. IEEE, 2002, pp. 496–502.

[5] C. Gosselin and L.-T. Schreiber, "Redundancy in parallel mechanisms: A review," *Applied Mechanics Reviews*, vol. 70, no. 1, p. 010802, 2018.

[6] M. Luces, J. K. Mills, and B. Benhabib, "A review of redundant parallel kinematic mechanisms," *Journal of Intelligent & Robotic Systems*, vol. 86, no. 2, pp. 175–198, 2017.

[7] J.-P. Merlet, "Redundant parallel manipulators," *Laboratory Robotics and Automation*, vol. 8, no. 1, pp. 17–24, 1996.

[8] J. Wang and C. M. Gosselin, "Kinematic analysis and design of kinematically redundant parallel mechanisms," *Journal of Mechanical Design*, vol. 126, no. 1, pp. 109–118, 2004.

[9] C. Gosselin, T. Laliberté, and A. Veillette, "Singularity-Free Kinematically Redundant Planar Parallel Mechanisms With Unlimited Rotational Capability," *IEEE Transactions on Robotics*, vol. 31, pp. 457–467, Mar. 2015.

[10] N. Baron, A. Philippides, and N. Rojas, "A novel kinematically redundant planar parallel robot manipulator with full rotatability," *Journal of Mechanisms and Robotics*, vol. 11, no. 1, 2019.

[11] H. Liao, T. Li, and X. Tang, "Singularity analysis of redundant parallel manipulators," in *2004 IEEE International Conference on Systems, Man and Cybernetics*, vol. 5. IEEE, 2004, pp. 4214–4220.

[12] I. Ebrahimi, J. A. Carretero, and R. Boudreau, "A family of kinematically redundant planar parallel manipulators," *Journal of Mechanical Design*, vol. 130, no. 6, p. 062306, 2008.

[13] L.-T. Schreiber and C. Gosselin, "Kinematically redundant planar parallel mechanisms: Kinematics, workspace and trajectory planning," *Mechanism and Machine Theory*, vol. 119, pp. 91–105, Jan. 2018.

[14] C. Gosselin and L.-T. Schreiber, "Kinematically redundant spatial parallel mechanisms for singularity avoidance and large orientational workspace," *IEEE Transactions on Robotics*, vol. 32, no. 2, pp. 286–300, 2016.

[15] D. Zlatanov, R. Fenton, and B. Benhabib, "A unifying framework for classification and interpretation of mechanism singularities," *Transactions-American Society of Mechanical Engineers Journal of Mechanical Design*, vol. 117, pp. 566–572, 1995.

[16] B. Hendrickson, "Conditions for Unique Graph Realizations," *SIAM Journal on Computing*, vol. 21, no. 1, pp. 65–84, 1992.

[17] L. Asimow and B. Roth, "The rigidity of graphs," *Transactions of the American Mathematical Society*, vol. 245, pp. 279–289, 1978.

[18] ——, "The rigidity of graphs, II," *Journal of Mathematical Analysis and Applications*, vol. 68, no. 1, pp. 171–190, 1979.

[19] J. Merlet, *Parallel Robots*. Dordrecht, The Netherlands: Springer Science & Business Media, 2006, ch. 2.

[20] N. Rojas and F. Thomas, "Forward kinematics of the general triple-arm robot using a distance-based formulation," in *Computational Kinematics*. Springer, 2018, pp. 257–264.

[21] N. Baron, A. Philippides, and N. Rojas, "A Geometric Method of Singularity Avoidance for Kinematically Redundant Planar Parallel Robots," in *International Symposium on Advances in Robot Kinematics*. Springer, 2018, pp. 187–194.

[22] H. Daniali, "Instantaneous center of rotation and singularities of planar parallel manipulators," *International Journal of Mechanical Engineering Education*, vol. 33, pp. 251–259, Jul. 2005.

[23] R. Di Gregorio, "A novel method for the singularity analysis of planar mechanisms with more than one degree of freedom," *Mechanism and Machine Theory*, vol. 44, no. 1, pp. 83–102, 2009.

[24] L.-T. Schreiber and C. Gosselin, "Exploiting the kinematic redundancy of a (6+3) degrees-of-freedom parallel mechanism," *Journal of Mechanisms and Robotics*, vol. 11, no. 2, p. 021005, 2019.